UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/779,466	02/16/2004	Matt Morris	350886-1023	5553

32914 7590 09/02/2009

GARDERE WYNNE SEWELL LLP
INTELLECTUAL PROPERTY SECTION
3000 THANKSGIVING TOWER
1601 ELM ST
DALLAS, TX 75201-4761

EXAMINER	
MAIS, MARK A	

ART UNIT	PAPER NUMBER
2419	

MAIL DATE	DELIVERY MODE
09/02/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No.	Applicant(s)	
	10/779,466	MORRIS, MATT	
	Examiner	Art Unit	
	MARK MAIS	2419	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *22 July 2009*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-3,5,7-12,14-17 and 32-38* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-3,5,7-12,14-17 and 32-38* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *16 February 2004* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☒ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☒ All b)☐ Some * c)☐ None of:

1.☒ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

Continued Examination Under 37 CFR 1.114

1. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on July 22, 2009 has been entered.

Claim Rejections - 35 USC § 112

2. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

3. Claim 1 is rejected under 35 U.S.C. 112, first paragraph, because the specification, while being enabling for an addressable storage means, does not reasonably provide enablement for arbitrarily addressable storage means. The specification does not enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the

invention commensurate in scope with these claims. Claims 2, 3, 5 and 7 are also rejected since they depend from claim 1 and contain the same deficiency. For examination purposes, the claims will be interpreted as addressable storage means.

4. Claim 8 is rejected under 35 U.S.C. 112, first paragraph, because the specification, while being enabling for an addressable storage means, does not reasonably provide enablement for arbitrarily addressable storage means. The specification does not enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the invention commensurate in scope with these claims. Claims 9 and 10 are also rejected since they depend from claim 1 and contain the same deficiency. For examination purposes, the claims will be interpreted as addressable storage means.

5. Claim 11 is rejected under 35 U.S.C. 112, first paragraph, because the specification, while being enabling for an addressable storage means, does not reasonably provide enablement for arbitrarily addressable storage means. The specification does not enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the invention commensurate in scope with these claims. Claims 12 and 14 are also rejected since they depend from claim 1 and contain the same deficiency. For examination purposes, the claims will be interpreted as addressable storage means.

Claim Rejections - 35 USC § 112

6. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

7. Claim 32 is rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention. Claim 32 depends from cancelled claim 6. There is insufficient antecedent basis for this limitation in the claim. Claim 33 is also rejected since it depends from claim 32 and contains the same deficiency.

Claim Rejections - 35 USC § 103

8. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Chapman et al. in view of Barnes et al. further in view of Gaudet et al. and Wegner et al.

9. Claims 1-3, 5, 7-12, 14-17 and 32-35 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chapman et al. (USP 6,304,552) in view of Barnes et al. (USP 7,382,787) further in view of Gaudet et al. (USP 6,421,348) and Wegner et al. (USP 6,032,192).

10. With regard to claims 1, 3, and 34-35, Chapman et al. discloses a stream routing unit **[lossy switch, Abstract]** for routing each of a plurality of input packet streams to any of a plurality of destinations, the stream routing unit comprising:

a plurality of input ports **[input ports, Abstract]** for receiving respective input streams;

a plurality of output ports **[output ports, Abstract]** associated with respective destinations to which the input packet streams can be routed;

storage means for holding packets of the input packet streams at *arbitrarily* addressable locations each identifiable by an address **[input buffers, col. 2, lines 4-5; the input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets, col. 7, lines 50-56]**;

an assignment data structure identifying for each *source of the* input *packet* stream at least one destination to which each input packet stream is to be routed **[switch fabric, col. 3, lines 34; col. 6, lines 6-21 (interpreted as a matrix–claim 3); the switch fabric is controlled by controller 308 which identifies each stream—using the mapping table—which is headed (routed) to a particular output port, col. 8, lines 14-24]**;

a packet allocation *table comprising an array including a plurality of slots* holding for each new incoming packet a source identifier identifying the *source* of the packet and the *arbitrary* address in the storage means where the packet is held, the packet allocation *table*

further *including a plurality of destination pointers, each destination pointer associated with one o f the output ports, each destination pointer being assignable to any slot so as to identify* the output ports associated with the intended destinations of a held packet, the *assigning of each destination pointer to a slot* being derived *using* the assignment data structure **[a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained below, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions. For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38). As a second example (col. 8, lines 38-57), by mapping incoming packets to egress queues/ports, it is well known that the memory is using write/read memory pointers (explained below); interpreted as the situation where the controller transmits HI priority packets first (availability through a specific port), then, if possible, transmits LO priority packets (availability on the same port or a different port), col. 10, lines 16-24 (the algorithm prioritizes HI priority packets over LO priority packets; it transmits HI priority packets in the order they are received—claim**

34); memory management for broadcasting/multicasting is known to those in the art. This prevents unnecessary multiple memory lookups with respect to output ports which output broadcast/multicast messages. For example, Wegner et al. discloses a queuing mechanism that withholds broadcast delivery (controls output from the memory) until all available ports can be used effectively (interpreted as becoming open—claim 35) and then submits messages to the ports simultaneously, col. 14, line 61 to col. 15, line 10. Thus, it would have been obvious to one of ordinary skill in the art at the time of the invention to have used a method of selective broadcasting/multicasting which uses a memory management scheme that limits memory lookups in order to provide simultaneous broadcast/multicast delivery]; *and*

processing means **[Fig. 3, controller 308]** *for controlling removal of packets from the storage means to the plurality of output ports using the destination pointers.*

The input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets **[Chapman et al., col. 7, lines 50-56]**. Chapman et al. further discloses that memory 310 implements queue mechanisms to accept data from the input ports **[col. 8, lines 11-24]**. Memory is merely the medium on which the buffers are created—the actual created [and deleted] and management of the buffers is controlled by controller 308 **[*See Id.*]**. Thus, buffers used in this type of memory management scheme use either (1) physical memory addresses or (2) pointers to the packet's location in the memory. Chapman et al. does not specifically teach the employment of such a memory management scheme for buffers 320/324. However, such a memory management scheme is well known to those in the art. For example, Barnes et al. (USP 7,382,787) discloses the use of a memory management scheme

capable of performing additional functions on packets within a router and then having to locate the packets for output processing using memory pointers, link lists, link list arrays, etc. **[col. 3, lines 8-22]**. Thus, it would have been obvious to use a memory management scheme disclosed in Barnes et al. on the FIFOs in Chapman et al. to perform packet functions and then be able to identify the packet by address/address pointer in order to perform read/write functions in order to transmit the packet stream in the correct order and to send the stream to the correct output port.

11. With regard to claim 2, the combination of references discloses that the input packet streams have a lower bit rate than output streams into which they are merged at the plurality of output ports **[Chapman et al., the input rates are much lower than the output rates, col. 2, lines 17-24; especially with bandwidth throttling mechanism, col. 9, lines 2-10; as well as sending HI and LO priority packets (col. 9, lines 28-39)]**.

12. With regard to claim 5, the combination of references discloses that the packet allocation *table* is associated with a write pointer which is configured to point to the next available slot in the array for the source identifier and address of the next incoming packet **[Chapman et al., a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; e.g., col. 8, lines 38-57; by mapping incoming packets to egress queues/ports, it is well known that the memory is using write/read memory pointers (as explained above); interpreted as the situation where the controller transmits HI priority packets first**

(through a specific port), then, if possible, transmits LO priority packets (the same port or a different port), col. 10, lines 16-24].

13. With regard to claims 32-33, the combination of references discloses an algorithm for destination pointer assignment and packet output from memory **[Chapman et al., a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38). For example (col. 8, lines 38-57), by mapping incoming packets to egress queues/ports, it is well known that the memory is using write/read memory pointers (as explained above); interpreted as the situation where the controller transmits HI priority packets first (availability through a specific port), then, if possible, transmits LO priority packets (availability on the same port or a different port), col. 10, lines 16-24 (the algorithm prioritizes HI priority packets over LO priority packets—claim 32; it transmits HI priority packets in the order they are received); memory management for broadcasting/multicasting is known to those in the art. This prevents unnecessary multiple memory lookups with respect to output ports which output broadcast/multicast messages. Wegner et al. discloses a queuing mechanism that withholds broadcast delivery (controls output from the memory—claim 33) until all available ports can be used effectively (interpreted as becoming open) and then submits messages to the ports simultaneously, col. 14, line 61 to col. 15, line 10]**.

14. With regard to claim 7, the combination of references discloses that the packets of a said input stream are of a common length **[Chapman et al., interpreted as common length IP packets, col. 5, lines 62-67]**.

15. With regard to claim 8, Chapman et al. discloses a data communication system **[network, col. 1, lines 36-38]** for routing incoming packets to at least one destination, the system comprising:

a plurality of packet stream sources each generating a packet stream **[multiple input sources for each input port (connects to other users, switches, network elements, col. 5, lines 61-62]**; a stream routing unit **[lossy switch, Abstract]** comprising:

a plurality of input ports **[input ports, Abstract]** for receiving respective input streams;

a plurality of output ports **[output ports, Abstract]** associated with respective destinations to which the input packet streams can be routed;

storage means for holding packets of the input packet streams at *arbitrarily* addressable locations each identifiable by an address **[input buffers, col. 2, lines 4-5; the input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets, col. 7, lines 50-56]**;

an assignment data structure identifying for each *source of the* input *packet* stream at least one destination to which each input packet stream is to be routed **[switch fabric, col. 3, lines 34; col. 6, lines 6-2; the switch fabric is controlled by controller 308**

which identifies each stream—using the mapping table—which is headed (routed) to a particular output port, col. 8, lines 14-24];

a packet allocation *table comprising an array including a plurality of slots* holding for each new incoming packet a source identifier identifying the *source* of the packet and the *arbitrary* address in the storage means where the packet is held, the packet allocation *table* further *including a plurality of destination pointers, each destination pointer associated with one of the output ports, each destination pointer being assignable to any slot as to identify* the output ports associated with the intended destinations of a held packet, the *assigning of each destination pointer to a slot* being derived *using* the assignment data structure **[a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained below, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions. For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-**

38). As a second example (col. 8, lines 38-57), by mapping incoming packets to egress queues/ports, it is well known that the memory is using write/read memory pointers (explained below); interpreted as the situation where the controller transmits HI priority packets first (availability through a specific port), then, if possible, transmits LO priority packets (availability on the same port or a different port), col. 10, lines 16-24 (the algorithm prioritizes HI priority packets over LO priority packets; it transmits HI priority packets in the order they are received); memory management for broadcasting/multicasting is known to those in the art. This prevents unnecessary multiple memory lookups with respect to output ports which output broadcast/multicast messages. For example, Wegner et al. discloses a queuing mechanism that withholds broadcast delivery (controls output from the memory) until all available ports can be used effectively (interpreted as becoming open) and then submits messages to the ports simultaneously, col. 14, line 61 to col. 15, line 10. Thus, it would have been obvious to one of ordinary skill in the art at the time of the invention to have used a method of selective broadcasting/multicasting which uses a memory management scheme that limits memory lookups in order to provide simultaneous broadcast/multicast delivery]; *and*

processing means **[Fig. 3, controller 308]** *for controlling removal of packets from the storage means to the plurality of output ports using the destination pointers;* and

a plurality of destinations for receiving packets of the packet streams generated by the sources **[multiple outputs for each output port (connects to other users, switches, network elements, col. 5, lines 61-62]**.

The input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets **[Chapman et al., col. 7, lines 50-56]**. Chapman et al. further discloses that memory 310 implements queue mechanisms to accept data from the input ports **[col. 8, lines 11-24]**. Memory is merely the medium on which the buffers are created—the actual created [and deleted] and management of the buffers is controlled by controller 308 ***[See Id.]***. Thus, buffers used in this type of memory management scheme use either (1) physical memory addresses or (2) pointers to the packet's location in the memory. Chapman et al. does not specifically teach the employment of such a memory management scheme for buffers 320/324. However, such a memory management scheme is well known to those in the art. For example, Barnes et al. (USP 7,382,787) discloses the use of a memory management scheme capable of performing additional functions on packets within a router and then having to locate the packets for output processing using memory pointers, link lists, link list arrays, etc. **[col. 3, lines 8-22]**. Thus, it would have been obvious to use a memory management scheme disclosed in Barnes et al. on the FIFOs in Chapman et al. to perform packet functions and then be able to identify the packet by address/address pointer in order to perform read/write functions in order to transmit the packet stream in the correct order and to send the stream to the correct output port.

16. With regard to claim 9, the combination of references discloses that at least one of the destinations comprises a programmable transport interface **[Chapman et al., interpreted as a repeater, col. 12, lines 15-28]**.

17. With regard to claim 10, the combination of references discloses that the input packet streams have a lower bit rate than output streams into which they are merged at the plurality of output ports **[Chapman et al., the input rates are much lower than the output rates, col. 2, lines17-24; especially with bandwidth throttling mechanism, col. 9, lines 2-10; as well as sending HI and LO priority packets (col. 9, lines 28-39)]**.

18. With regard to claim 11, Chapman et al. discloses a method of routing packet streams **[Abstract]** from a plurality of sources to any of a plurality of destinations, the method comprising:

receiving said packet streams **[multiple input sources for each input port (connects to other users, switches, network elements, col. 5, lines 61-62]**;

identifying for each input packet stream at least one destination to which each input packet stream is to be routed using an assignment data structure **[a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained below, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to**

perform these functions. For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38)];

holding each packet of the packet stream *in a storage means* at an addressable location identifiable by an address in *that* storage means **[input buffers, col. 2, lines 4-5; the input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets, col. 7, lines 50-56]**;

holding for each new incoming packet *a packet allocation data structure which stores* a source identifier identifying the origin of the packet and the address in the storage means where the packet is held [**a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained below, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions. For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38)]**;

holding information *in an assignment data structure* identifying the intended destination of the packet derived from the assignment data structure **[a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained below, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions. For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38)]**;

using said *assignment data structure* information identifying the intended destination to *further include in the packet allocation data structure information identifying output ports associated with intended destinations of the held packet; routing* the packet from the storage means to the or each output port **[output ports, Abstract]** associated with the respective identified destination(s) **[a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained below, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is**

allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions. For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38). As a second example (col. 8, lines 38-57), by mapping incoming packets to egress queues/ports, it is well known that the memory is using write/read memory pointers (explained below); interpreted as the situation where the controller transmits HI priority packets first (availability through a specific port), then, if possible, transmits LO priority packets (availability on the same port or a different port), col. 10, lines 16-24 (the algorithm prioritizes HI priority packets over LO priority packets; it transmits HI priority packets in the order they are received); memory management for broadcasting/multicasting is known to those in the art. This prevents unnecessary multiple memory lookups with respect to output ports which output broadcast/multicast messages. For example, Wegner et al. discloses a queuing mechanism that withholds broadcast delivery (controls output from the memory) until all available ports can be used effectively (interpreted as becoming open) and then submits messages to the ports simultaneously, col. 14, line 61 to col. 15, line 10. Thus, it would have been obvious to one of ordinary skill in the art at the time of the invention to have used a method of selective broadcasting/multicasting which uses a

memory management scheme that limits memory lookups in order to provide simultaneous broadcast/multicast delivery].

The input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets **[Chapman et al., col. 7, lines 50-56]**. Chapman et al. further discloses that memory 310 implements queue mechanisms to accept data from the input ports **[col. 8, lines 11-24]**. Memory is merely the medium on which the buffers are created—the actual created [and deleted] and management of the buffers is controlled by controller 308 [***See Id.***]. Thus, buffers used in this type of memory management scheme use either (1) physical memory addresses or (2) pointers to the packet's location in the memory. Chapman et al. does not specifically teach the employment of such a memory management scheme for buffers 320/324. However, such a memory management scheme is well known to those in the art. For example, Barnes et al. (USP 7,382,787) discloses the use of a memory management scheme capable of performing additional functions on packets within a router and then having to locate the packets for output processing using memory pointers, link lists, link list arrays, etc. **[col. 3, lines 8-22]**. Thus, it would have been obvious to use a memory management scheme disclosed in Barnes et al. on the FIFOs in Chapman et al. to perform packet functions and then be able to identify the packet by address/address pointer in order to perform read/write functions in order to transmit the packet stream in the correct order and to send the stream to the correct output port.

19. With regard to claim 12, the combination of references discloses that the input packet streams have a lower bit rate than output streams into which they are merged at the output ports **[Chapman et al., the input rates are much lower than the output rates, col. 2, lines 17-24;**

especially with bandwidth throttling mechanism, col. 9, lines 2-10; as well as sending HI and LO priority packets (col. 9, lines 28-39)].

20. With regard to claim 14, the combination of references discloses associating each slot with a write pointer which is configured to point to the next available slot in the array for the source identifier and address of the next incoming packet **[Chapman et al., a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; for example (col. 8, lines 38-57), by mapping incoming packets to egress queues/ports, it is well known that the memory is using write/read memory pointers (explained in the rejection of claim 11 above); for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions (explained in the rejection of claim 11 above). For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38); this is interpreted as the situation where the controller transmits HI priority packets first (availability through a specific port), then, if possible, transmits LO priority packets (availability on the same port or a different port), col. 10, lines 16-24 (the algorithm**

prioritizes H1 priority packets over LO priority packets; it transmits H1 priority packets in the order they are received)].

Chapman et al. in view of Barnes et al. and Gaudet et al.

21. Claims 15-23 and 36-38 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chapman et al. (USP 6,304,552) in view of Barnes et al. (USP 7,382,787) and Gaudet et al. (USP 6,421,348).

22. With regard to claim 15, Chapman et al. discloses a device **[lossy switch, Abstract]** for delivering incoming packets to at least one destination **[Abstract]**, the device comprising:

an addressable memory which stores incoming packets at a plurality of address locations in the memory **[input buffers, col. 2, lines 4-5; the input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets, col. 7, lines 50-56]**;

a source to destination matrix for mapping at least one source to at least one destination **[switch fabric, col. 3, lines 34; col. 6, lines 6-21; interpreted as a matrix; the switch fabric is controlled by controller 308 which identifies each stream—using the mapping table—which is headed (routed) to a particular output port, col. 8, lines 14-24]**;

a packet allocation table *including a plurality of slots, each slot* associating a source for a particular packet with the address location in the addressable a memory **[input buffers, col. 2,**

lines 4-5; the input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets, col. 7, lines 50-56] *where* the particular packet is stored; *a plurality of destination pointers associated with the packet allocation table, each destination pointer having an associated destination, and each destination pointer being assignable to any slot in the packet allocation table* **[a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained below, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions. For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38)]** and

an algorithm for controlling removal of the incoming packets from a memory to at least one destination **[packet discard, col. 11, lines 58-63]**, wherein the incoming packets have a lower bit-rate than packets delivered to the at least one destination **[the input rates are much lower than the output rates, col. 2, lines 17-24; especially with bandwidth throttling**

mechanism, col. 9, lines 2-10; as well as sending HI and LO priority packets (col. 9, lines 28-39)].

The input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets **[Chapman et al., col. 7, lines 50-56]**. Chapman et al. further discloses that memory 310 implements queue mechanisms to accept data from the input ports **[col. 8, lines 11-24]**. Memory is merely the medium on which the buffers are created—the actual created [and deleted] and management of the buffers is controlled by controller 308 **[*See Id.*]**. Thus, buffers used in this type of memory management scheme use either (1) physical memory addresses or (2) pointers to the packet's location in the memory. Chapman et al. does not specifically teach the employment of such a memory management scheme for buffers 320/324. However, such a memory management scheme is well known to those in the art. For example, Barnes et al. (USP 7,382,787) discloses the use of a memory management scheme capable of performing additional functions on packets within a router and then having to locate the packets for output processing using memory pointers, link lists, link list arrays, etc. **[col. 3, lines 8-22]**. Thus, it would have been obvious to use a memory management scheme disclosed in Barnes et al. on the FIFOs in Chapman et al. to perform packet functions and then be able to identify the packet by address/address pointer in order to perform read/write functions in order to transmit the packet stream in the correct order and to send the stream to the correct output port.

23. With regard to claim 16, the combination of references discloses a memory for holding the incoming packets at addressable locations each identifiable by an address **[Chapman et al., FIFO buffers, col. 7, line 50 to col. 8, line 10; the input buffers are memory spaces which**

hold incoming data packets until a routing decision is made as to handle the packets, col. 7, lines 50-56].

Chapman et al. further discloses that memory 310 implements queue mechanisms to accept data from the input ports **[col. 8, lines 11-24]**. Memory is merely the medium on which the buffers are created—the actual created [and deleted] and management of the buffers is controlled by controller 308 **[*See Id.*]**. Thus, buffers used in this type of memory management scheme use either (1) physical memory addresses or (2) pointers to the packet's location in the memory. Chapman et al. does not specifically teach the employment of such a memory management scheme for buffers 320/324. However, such a memory management scheme is well known to those in the art. For example, Barnes et al. (USP 7,382,787) discloses the use of a memory management scheme capable of performing additional functions on packets within a router and then having to locate the packets for output processing using memory pointers, link lists, link list arrays, etc. **[col. 3, lines 8-22]**. Thus, it would have been obvious to use a memory management scheme disclosed in Barnes et al. on the FIFOs in Chapman et al. to perform packet functions and then be able to identify the packet by address/address pointer in order to perform read/write functions in order to transmit the packet stream in the correct order and to send the stream to the correct output port. .

24. With regard to claim 17, the combination of references discloses a plurality of input ports for receiving respective input packets **[Chapman et al., multiple input sources for each input port (connects to other users, switches, network elements, col. 5, lines 61-62]**; and

a plurality of output ports associated with respective destinations to which the input packets can be routed **[Chapman et al., multiple outputs for each output port (connects to other users, switches, network elements, col. 5, lines 61-62]**.

25. With regard to claim 36, Chapman et al. discloses a stream routing unit, comprising:

a plurality of input ports **[input ports, Abstract]**, each input port receiving an input packet stream;

a plurality of output ports **[output ports, Abstract]**, each output port outputting an output packet stream;

a memory including a plurality of addressable memory locations **[input buffers, col. 2, lines 4-5; the input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets, col. 7, lines 50-56]**;

a source-to-destination matrix *mapping* each source of the input packet streams coupled to the input ports *to* one or more destinations, for packets within those input packet streams, which are coupled to receive the output packet streams from the output ports **[a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained below, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node**

addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions (explained below). For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38)];

a processor for storing packets of the input packet streams in the memory and for retrieving stored packets from the memory to form the output packet streams **[Fig. 3, Controller 308]**;

the processor filling a packet allocation table which includes a plurality of slot locations, each slot location storing a source identifier which identifies a source of the received packet stream to which a given packet belongs linked in the *slot of the* packet allocation table to an address in the memory for the addressable memory location where that given packet has been stored by the processor **[Controller 308 runs a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained below, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions**

(explained below). For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38)];

a destination pointer, associated with each one of the output ports, implemented by the processor for pointing to a slot location in the packet allocation table from which the address of the given packet is retrieved, the destination pointer pointing to the slot location *when the source identifier in the slot is associated with a source that is mapped through* the source-to-destination matrix *to a destination coupled to an output port and that output port is associated with that destination pointer* **[Controller 308 runs a routing table (w/plural slots) for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained below, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions (explained below). For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38)]**;

the processor retrieving the given packet from the memory at the address provided in the slot location pointed at by the destination pointer, sending the retrieved given packet to each output port associated with the destination that is linked in the source-to-destination matrix with the source identified in the slot location for inclusion in the output packet stream of the output port **[Controller 308 runs a routing table for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained below, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions (explained below). For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38)]**.

The input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets **[Chapman et al., col. 7, lines 50-56]**. Chapman et al. further discloses that memory 310 implements queue mechanisms to accept data from the input ports **[col. 8, lines 11-24]**. Memory is merely the medium on which the buffers are created—the actual created [and deleted] and management of the buffers is controlled by controller 308 **[*See Id.*]**. Thus, buffers used in this type of memory management scheme use either (1) physical

memory addresses or (2) pointers to the packet's location in the memory. Chapman et al. does not specifically teach the employment of such a memory management scheme for buffers 320/324. However, such a memory management scheme is well known to those in the art. For example, Barnes et al. (USP 7,382,787) discloses the use of a memory management scheme capable of performing additional functions on packets within a router and then having to locate the packets for output processing using memory pointers, link lists, link list arrays, etc. **[col. 3, lines 8-22]**. Thus, it would have been obvious to use a memory management scheme disclosed in Barnes et al. on the FIFOs in Chapman et al. to perform packet functions and then be able to identify the packet by address/address pointer in order to perform read/write functions in order to transmit the packet stream in the correct order and to send the stream to the correct output port.

26. With regard to claim 37, the combination of references discloses a write pointer implemented by the processing means for pointing to an open slot location in the packet allocation table to which the source identifier and address of the given packet are written **[Chapman et al., Controller 308 runs a routing table for mapping destination address of incoming packets to the output port, col. 7, lines 65-67; it holds the source address, col. 10, lines 5-12; col. 4, lines 28-32; as explained in the rejection of claim 36, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers; for example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet**

to the proper access point router, col. 10, lines 50-63; thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions (explained in the rejection of claim 36 above). For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38)].

27. With regard to claim 38, the combination of references discloses that a bit rate of the input packet streams is lower than a bit rate of the output packet streams **[Chapman et al., the input rates are much lower than the output rates, col. 2, lines 17-24; especially with bandwidth throttling mechanism, col. 9, lines 2-10; as well as sending HI and LO priority packets (col. 9, lines 28-39)]**.

Response to Arguments

28. Applicant's arguments filed on July 22, 2009 have been fully considered but they are not persuasive.

29. With respect to claim 1, Applicants state that Chapman et al. fails to disclose a storage means which is "arbitrarily addressable" **[See Applicants' Amendment dated July 22, 2009, page 11, paragraph 3]**. The examiner respectfully disagrees.

30. As indicated above in the rejections of independent claims 1, 8 and 11, those claims are rejected under 35 U.S.C. 112, first paragraph, because the specification, while being enabling for an addressable storage means, does not reasonably provide enablement for arbitrarily addressable storage means. The specification does not enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the invention commensurate in scope with these claims. The dependent claims are also rejected since they depend from the independent claims and contain the same deficiency. For examination purposes, the claims will be interpreted as addressable storage means.

31. With respect to claim 1, Applicants state that the Chapman fails to disclose a "a packet allocation table…[with] a plurality of slots [and]…a plurality of destination pointer[s]…being assignable to any slot" **[See Applicants' Amendment dated July 22, 2009, page 11, paragraph 4]**. The examiner respectfully agrees. However, the examiner respectfully disagrees with the argument that the combination of references fails to disclose, teach, or suggest the recited claim limitations **[See Applicants' Amendment dated July 22, 2009, page 11, paragraph 4 to page 12, paragraph 1]**. Applicants make similar arguments with respect to claims 8, 11, 15, and 36 **[See Applicants' Amendment dated July 22, 2009, page 13, paragraphs 4-7]**.

32. Chapman et al. discloses a routing table for mapping destination address of incoming packets to the output port **[col. 7, lines 65-67]**. It holds the source address **[col. 10, lines 5-12; col. 4, lines 28-32]**. Moreover, it is well known in a memory-mapping scheme that the

location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers. For example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router **[col. 10, lines 50-63]**. Thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions. For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38). As a second example (col. 8, lines 38-57), by mapping incoming packets to egress queues/ports, it is well known that the memory is using write/read memory pointers. This is interpreted as the situation where the controller transmits HI priority packets first (availability through a specific port), then, if possible, transmits LO priority packets (availability on the same port or a different port) **[col. 10, lines 16-24; (the algorithm prioritizes HI priority packets over LO priority packets; it transmits HI priority packets in the order they are received)]**.

The input buffers are memory spaces which hold incoming data packets until a routing decision is made as to handle the packets **[Chapman et al., col. 7, lines 50-56]**. Chapman et al. further discloses that memory 310 implements queue mechanisms to accept data from the input ports **[col. 8, lines 11-24]**. Memory is merely the medium on which the buffers are created—the actual created [and deleted] and management of the buffers is controlled by controller 308 [***See Id.***]. Thus, buffers used in this type of memory management scheme use either (1) physical memory addresses or (2) pointers to the packet's location in the memory. Chapman et al. does

not specifically teach the employment of such a memory management scheme for buffers 320/324. However, such a memory management scheme is well known to those in the art. For example, Barnes et al. (USP 7,382,787) discloses the use of a memory management scheme capable of performing additional functions on packets within a router and then having to locate the packets for output processing using memory pointers, link lists, link list arrays, etc. **[col. 3, lines 8-22]**. Thus, it would have been obvious to use a memory management scheme disclosed in Barnes et al. on the FIFOs in Chapman et al. to perform packet functions and then be able to identify the packet by address/address pointer in order to perform read/write functions in order to transmit the packet stream in the correct order and to send the stream to the correct output port.

33. With respect to claim 1, Applicants argue that Gaudet et al. does not teach the source information and a corresponding storage address **[See Applicants' Amendment dated July 22, 2009, page 12, paragraph 4 to page 13, paragraph 1]**. The examiner respectfully disagrees.

34. First, as noted in the rejection of claim 1 above, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer **[col. 7, lines 28-38]**.

35. Second, in response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on

combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.,* 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

36. With respect to claim 1, Applicants state that Wegner fails to teach a "packet allocation data table" **[See Applicants' Amendment dated July 22, 2009, page 12, paragraph 2]**. The examiner respectfully agrees. However, the examiner disagrees with the apparent argument that the combination of references fails to disclose, teach, or suggest the recited claim limitations **[See Applicants' Amendment dated July 22, 2009, page 13, paragraph 2]**.

37. First, as noted in the rejection of claim 1 above, memory management for broadcasting/multicasting is known to those in the art. This prevents unnecessary multiple memory lookups with respect to output ports which output broadcast/multicast messages. For example, Wegner et al. discloses a queuing mechanism that withholds broadcast delivery (controls output from the memory) until all available ports can be used effectively (interpreted as becoming open) and then submits messages to the ports simultaneously **[col. 14, line 61 to col. 15, line 10]**. Thus, it would have been obvious to one of ordinary skill in the art at the time of the invention to have used a method of selective broadcasting/multicasting which uses a memory management scheme that limits memory lookups in order to provide simultaneous broadcast/multicast delivery.

38. Second, in response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on

combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.,* 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

39. With respect to claim 36, Applicants argue, apparently, that Chapman et al. fails to disclose a "a source-to-destination matrix mapping each source of the input packet streams coupled to the input ports to one or more destinations, for packets within those input packet streams, which are coupled to receive output packet streams from the output ports" **[See Applicants' Amendment dated July 22, 2009, page 13, paragraph 8]**. The examiner respectfully agrees. However, the examiner respectfully disagrees with the argument that the combination of references fails to disclose, teach, or suggest the recited claim limitations **[See Applicants' Amendment dated July 22, 2009, page 13, paragraph 8 to page 14, paragraph 3]**.

40. First, Chapman et al. discloses a routing table for mapping destination address of incoming packets to the output port **[col. 7, lines 65-67]**. It holds the source address **[col. 10, lines 5-12; col. 4, lines 28-32]**. As explained in the rejection of claim 36 above, it is well known in a memory-mapping scheme that the location/address (within the buffer/queue) of the incoming packet is allocated/used in conjunction with write/read pointers. For example, the destination is read from the header of the packet, the routing table indicates the packets' delivery address, then the switches' input and output ports are used to derive the entry (source identifier) and exit (destination) transport node addresses which deliver the packet to the proper access point router **[col. 10, lines 50-63]**. Thus, a memory address or memory pointer (where the packet is held) must necessarily be used to perform these functions (explained in the rejection of claim 36

above). For instance, Gaudet et al. (USP 6,421,348), discloses a data exchanger which monitors the Source ID and places cells in identified/identifiable memory locations of an associated buffer (col. 7, lines 28-38).

41. Second, in response to applicant's arguments against the references individually, one cannot show nonobviousness by attacking references individually where the rejections are based on combinations of references. See *In re Keller*, 642 F.2d 413, 208 USPQ 871 (CCPA 1981); *In re Merck & Co.,* 800 F.2d 1091, 231 USPQ 375 (Fed. Cir. 1986).

42. With respect to claim 36, Applicants state that Chapman et al. fails to disclose holding a "source address." The examiner respectfully disagrees.

43. As noted in the rejection of claim 1 above, Chapman et al. discloses holding the source address **[col. 10, lines 5-12]**. More specifically, that citation discusses the source input queue. The source input queue stores the source address **[col. 4, lines 28-32]**. Then queue congestion is checked **[*See Id.*]**.

Conclusion

44. Any inquiry concerning this communication or earlier communications from the examiner should be directed to MARK MAIS whose telephone number is (571)272-3138. The examiner can normally be reached on 5am-4pm.

45. If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Pankaj Kumar can be reached on 571-272-3011. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

46. Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

August 13, 2009

/Mark A. Mais/

Examiner, Group Art Unit 2419

/Pankaj Kumar/

Supervisory Patent Examiner, Art Unit 2419